



**Fermi National Accelerator Laboratory**

## **D0 Silicon Strip Detector Upgrade Project**

### **STAND-ALONE SVX SEQUENCER BOARD**

**D0 Engineering Note Number 3823.110-EN-504**

12/9/98

M. Utes

---

1	INTRODUCTION .....	2
2	I/O CONFIGURATION .....	2
2.1	General Crate Configuration .....	2
2.2	Interface to the SVX chips.....	3
2.3	Data Readout .....	3
2.4	VME.....	5
3	GENERAL CONTROL CIRCUITRY .....	5
3.1	Main Control .....	5
3.2	Control/Status Register.....	5
4	DIAGNOSTICS .....	6
4.1	Snapshot Register.....	6
4.2	FIFO Test Register .....	7
4.3	Front Panel .....	7
5	VME REGISTERS.....	8
6	OPERATION .....	8
6.1	Setting the Base Address and Interrupt Level.....	8
6.2	Initialization.....	8
6.3	Running Data Acquisition Cycles .....	9
6.4	Reading the Data .....	11
6.5	Error Recovery .....	11
7	JTAG.....	12
APPENDICES		
A	Schematic Diagrams. ....	12
B	EPLD Files. ....	12

## 1 INTRODUCTION

The Stand-alone Sequencer boards are 6U by 160mm VME-based circuit boards that will be used to test silicon detectors without having to assemble complete data acquisition systems; they interface directly to HDIs via interface boards, and read out the data via the VME backplane to a BIT-3 board or a Motorola 68K. The basic function is to control the SVX chips for data acquisition and when a data cycle is requested, to gather the SVX data and relay the data to the processor in the crate. Functions and features are as follows:

- Initialization of two SVX chip strings using the VME write cycles.
- Real time manipulation of the SVX control lines to effect data acquisition, digitization, and readout based on the NRZ/Clock signals from the “Controller” circuit built into the board.
- Storage of SVX readout data in an on-board FIFO memory.
- Identification header and end of data trailer tacked onto data stream.
- VME register which can read the current values of the control and data lines.
- VME registers for crossing pulse width, calibration pulse voltage, and calibration pipeline select.
- VME-writeable register for triggering a laser, etc. followed by an acquisition cycle.
- TTL front panel input to trigger an acquisition cycle, e.g. from a scintillator.
- Synch Trig, Veto, Busy and Preamp Reset TTL outputs on front panel LEMOs.
- On-board 53.104 MHz oscillator.
- VME-triggerable Cal-inject cycle.
- A VME interrupt which will notify the crate processor that data exists to be read out.
- Front-panel connections to one 3M gray cable (two HDIs).
- Front panel displays and LEDs show the board status at a glance.
- In-system programmable EPLDs are programmed via Altera’s “Bitblaster”.

## 2 I/O CONFIGURATION

### 2.1 General Crate Configuration

The boards will reside in any 6U by 160mm VME crate. Slot one in each crate will house a VME Master/Controller, most likely a BIT-3 card which interfaces the crate to a PC. In this configuration, an Excel spreadsheet running visual basic commands will cause VME cycles on the backplane to initialize the board, to download the SVX chips, to initiate data acquisition cycles, and to read the SVX data back into the spreadsheet for analysis.

## 2.2 Interface to the SVX Chips

A single 3M Mini-D connector is the interface to the two SVX chip strings, and accepts the 3M cable that goes to a Transition board connected to two HDIs. The bidirectional bus signals use transceiver chips to drive the 82 $\Omega$  cable. Other control signals use TTL drivers. A direction line is routed to the Transition board to control the transceivers on that board. Another signal called HDI-EN is sent to the Transition board to tell it to turn power on to the SVX chips; it is an optional signal.

## 2.3 Data Readout via VME

During readout of the SVX chips, the eight data bits from each of two HDIs is strobed into FIFO memories. These sixteen bits are readable as one register where each byte of the 16-bit word represents one SVX chip string. Since the data is stored in a FIFO, reading the data is accomplished by continuously reading one VME register a number of successive times. The data for each HDI is arranged as follows:

**Sequencer ID**  
**HDI # & Status**  
**Chip ID**  
**Byte of zeroes**  
**Channel ID**  
**Data**  
**Channel ID**  
**Data**  
**...and so on**  
**hex "C0" (for end marker)**

The availability of data in the FIFO is signalled by a VME interrupt. The interrupt level is selectable via 3 dipswitches and a jumper.

There is an Interrupt Control Register (1A) which uses three bits to handle interrupt related functions. The Interrupt Enable bit, when set to 1, allows the generation of interrupt requests when a new data event is received or when an interrupt request is forced by the Interrupt Reset bit. This bit powers-up in the disabled state. See Table 2.3.

The Interrupt Reset bit is a read/write bit which, when read, indicates the state of the interrupt request: 0 when there is no interrupt requested and 1 when an interrupt request is in progress. When this register is written with a 0, the interrupt request is dropped and when written with a 1 (assuming it was already 0) will cause an interrupt request transaction just as if an event had been received. This bit powers-up to 0.

The Event Veto bit is a read/write bit which, in the 1 state, disallows the processing of further events. The processor sets and clears this bit.

The sequence of interactions between the processor and the S.A. Sequencer are as follows:

- 1) At initialization:
  - a) The processor loads an interrupt vector address into the register (see section 5).
  - b) The processor sets the Interrupt Enable/Disable signal to the enable state.
- 2) For a new event:
  - a) The S.A. Sequencer requests an interrupt of the processor, supplying the Interrupt Vector Address on the VME bus during the transaction.
  - b) The processor enters the interrupt service routine, sets the Event Veto signal, clears the Interrupt Request signal and exits the interrupt service routine.
  - c) When the processor has finished reading data from the FIFO it clears the Event Veto signal, enabling the processing of further events.

### **2.3.1 Header in Data Readout**

The Sequencer ID is a number from 0 to 255. The next byte is defined, starting with the MSB, as follows:

**Downloaded status bit 2**  
**Downloaded status bit 1**  
**Downloaded status bit 0**  
**Control Link No\_Synch**  
**Control Link Parity Error**  
**Zero**  
**Zero**  
**HDI identification**

The Downloaded status bits are provided for general use and are set via the VME Status register.

The No\_Synch and Par\_Err bits are reset to zero shortly after the data header is strobed into the FIFO. If a constant or momentary fault condition occurs after this time, the appropriate bit will be set to one and latched for readout during the next event readout.

### **2.3.2 Trailer in Data Readout**

The Trailer for each HDI data stream will be hex C0 and will be appended to the data immediately following the last data byte of the last chip on its HDI. The forced readout of channel 127 of the last chip on each HDI (a feature of the SVX2E chip) will not be used; the option to use this feature, however, still exists.

## 2.4 VME

The VME address range of the board is 50Dxxx. Bits A09 through A05 are selectable via dipswitches (on = 0) to accommodate having multiple cards in one crate. The module is an A24/D16 slave. Interrupt level is selectable via three dipswitches (on = 0) and a jumper.

## 3 GENERAL CONTROL CIRCUITRY

### 3.1 Main Control

The NRZ control link signal from U89 is the main input to the Control EPLD (U67). NRZ carries codes like Acquire, Digitize, Readout, etc. and these codes change based on either VME commands or front panel trigger signals. U67 manipulates the SVX bus lines, Mode lines, and TNBR in the appropriate fashion based on the commands received over the Control link.

The SVX chips can be initialized when NRZ is sending the Idle command. Data taking commences when the Acquire code is sent. Occasional Preamplifier Reset codes, about one every 21 us, will keep the SVX preamps from saturating during the acquire mode. When a trigger occurs, or a Cal-Inject command is received, the Trigger code is sent which causes the Sequencer to effect digitization in the SVX chips. After a predetermined delay, U89 sends the Readout code, and the chips are instructed to read out the data.

### 3.2 Control/Status Register

The utility EPLD (U125) operates the Sequencer's Status Register (subaddress 16). The bits are defined in Table 2 and Table 3.

<u>Bit</u>	<u>Function</u>
0	Bit appears in Status Word at beginning of readout
1	Bit appears in Status Word at beginning of
2	Bit appears in Status Word at beginning of readout MSB
3	High if a Parity Error has occurred in the Control Link
4	High if the Control Link had lost synch.
5	Write high to load AF flag offset
6	Selects HDI to appear in diagnostics
7	High = general Reset to board
8	High if Trig code is on NRZ
9	High if Cal_Inj code is on NRZ
10	High if Acquire code is on NRZ
11	Reserved
12	Reserved
13	Reserved
14	Reserved
15	Reserved

Table 2.1 Control/Status Register (0E) bits

<u>Bit</u>	<u>Function</u>
0	Pulse Shadow Register for HDI 0
1	Pulse Shadow Register for HDI 1
2	Unused
3	Unused
4	Force SVX chips to Idle state
5-15	Unused

Table 2.2 Shadow Register (18) bits

<u>Bit</u>	<u>Function</u>
0	Interrupt Enable
1	Interrupt Reset
2	Event Veto
3-15	Unused

Table 2.3 Interrupt Control Register (1A) bits

<u>Bit</u>	<u>Function</u>
0	Enable Power to HDI 1
1	Enable Power to HDI 2
2	Unused
3	Unused
4	Ignore HDI 1
5	Ignore HDI 2

Table 2.4 HDI Control Register (12) bits

## 4 DIAGNOSTICS

### 4.1 Snapshot Register

Built-in diagnostics are useful to isolate the cause and location of a malfunction. If the board freezes and no control is possible, reading this register will give the current status of the bus lines, Mode 0 and Mode 1, TNBR, Nosynch, and the four BNBR lines. This is called the “Snapshot” register. Bit 6 of the Status Register chooses which of the four HDIs will appear for the Bus lines. Table 4 shows the bit arrangement.

<u>Bit</u>	<u>Signal</u>
0-7	Bus 0-7
8	TNBR
9	Mode 1
10	Mode 0
11	NoSynch
12	BNBR1
13	BNBR2
14	BNBR3
15	BNBR4

Table 4. Bit assignment for diagnostic register (14).

## 4.2 FIFO Test Register

Data can be written into register 16 up to 4096 times and then read out via the data readout register as a test of the FIFO.

## 4.3 Front Panel

Front panel displays include two hex displays of the eight bus lines and one hex display of the current NRZ code. LEDs indicate Idle, Cal\_Inj cycle, Digitize cycle, External Trigger, FIFO reads, Parity error, No Synch, VME strobe. Sixteen SVX control and bus lines are available for probing with an oscilloscope or logic analyzer.

### 4.3.1 Display Showing the NRZ Code

The front panel display that shows the current NRZ code can be interpreted by the following:

<u>Display</u>	<u>Code</u>
0	Idle for Initialize
1	Acquire
3	Trigger
A	Ramp
2	Digitize
6	Readout
5	Preamp Reset
7	Cal_Inject
Others	Unused

### 4.3.2 Display Showing the SVX Bus Lines

The two hex displays that show the current value of the bus lines going to the SVX is located directly underneath the NRZ display. The two digit number represents the eight bus (control) lines, and the MSB is bus 7. Thus, referring to the SVX Beginner's Guide, Acquire mode will have a value of 7C and Idle mode will have 71.

### 4.3.3 LEDs and LEMO Connectors

See Appendix C for a drawing of the front panel.



## 5 VME REGISTERS

The following table defines the VME registers used by the Stand-Alone Sequencer. The address shown is for a board with dipswitches A09-A05 set at zero.

<u>Address</u>	<u>Function</u>
50D000	Initialize String 1
50D002	Initialize String 2
50D004	Data Readout Register
50D006	Crossing Pulse Width
50D008	Calibration Voltage
50D00A	Calibration Pipeline
50D00C	Sequencer ID (appears in readout data stream)
50D00E	Control/Status Register
50D010	Shadow pulse register after initializing SVXs
50D012	HDI Control Register
50D014	Read current state of bus and control lines
50D016	Register for testing FIFO in data path
50D018	Triggering Register
50D01A	Control/Status Register for Interrupts
50D01C	Sets the Interrupt-Vector ID
50D01E	Spare

Table 5. Definitions of the VME registers.

## 6 Operation

### 6.1 Setting the Base Address and Interrupt Level

Each board in a crate must have a unique VME base address, which is set by the top five segments of the eight-fold dipswitch on the board. The switch nearest the top corresponds to VME address bit A05, the second corresponds to A06, the third to A07, the fourth to A08 and the fifth to A09. Setting a switch to ON sets the bit to 0. Thus, with all five ON, the base address is 50D000, with only the top switch OFF the base address is 50D020, with only the second switch OFF the base address is 50D040, with only the top two switches OFF the base address is 50D060, and so on.

The other three switches select the interrupt level on which the board operates; the LSB is nearest the top. A jumper must be placed on the corresponding pair JP1 through JP7.

### 6.2 Initialization

The following is a list of a general initial setup of the SVX Sequencer. Refer to tables 2 and 5. Set the crossing pulse width to h25, Cal voltage to h80, Cal pipeline to match what is downloaded into the SVX chips. Load the Sequencer ID with a value corresponding to some meaningful relationship with the detector configuration. If there is only one HDI being used, set

the ignore bit as shown in table 2.4. Set bits 7 and 5 in the Control/Status register (table 2.1); bits 0 through 3 are user-defined. Write h80 to the Trigger register to prepare for SVX initialization.

To initialize the chips, bit D00 of each of the initialization registers is used for the serial download to the chips. Therefore, one bit propagates through the chip-string for every VME cycle. Each chip has 190 bits in its initialization shift register. See the “Beginner’s Guide to SVX II” for details on the initialization bits.

After an HDI is downloaded, a “1” must be written to bits 0 and 1 in the Shadow Register in order to latch the critical bits into the SVX chips. After this fleeting bit is sent, the SVX input shift register may be read to verify a correct download. Read bit D04 of the initialization registers to verify the download.

### **6.3 Running Data Acquisition Cycles**

#### **TRIG-IN Front Panel Input operation**

A high pulse applied to the TRIG-IN input will cause a data acquisition cycle to occur in the SVX chip. Typically, this pulse is applied to the Sequencer at the same time charge is injected into the SVX. The charge pulse should be synchronized to the SVX crossing clock using the VETO-OUT signal.

#### **VETO-OUT Front Panel Output operation**

VETO-OUT is provided to be externally ANDed with a random (e.g. scintillator) trigger so that a charge pulse is applied to the SVX at the proper time. The idea is to allow the charge pulse to arrive at the SVX within a window of time that begins at the falling edge of the SVX crossing clock and lasts for approximately 56ns. This narrow window assures that none of the pulse’s charge gets injected into the subsequent pipeline cell of the SVX chip. A pulse concurrent with the charge pulse should be applied to the TRIG-IN input to trigger the SVX digitization cycle.

VETO-OUT is synchronous with the clock and pulses high for 56ns in 132ns crossing mode, beginning 58ns after the falling edge of the SVX acquire clock. It pulses high for 130ns in 396ns crossing mode, beginning 22ns after the falling edge of the SVX acquire clock.

A user-provided delay on the charge pulse must be provided to ensure the earliest moment charge can be injected is just after the falling edge of the acquire clock as measured at the SVX. VETO-OUT is low during preamp resets and while BUSY is low (while the SVX is in digitize and readout mode).

#### **SYNCTRIG output operation**

This output signal is designed to be used to synchronously trigger an external pulse into the SVX chip (e.g. a laser). When bit D01 is written high to the Trigger Register, a pulse will appear on this output, and a digitization/readout cycle will commence. In 132ns mode, this pulse occurs in pipeline 5, 130ns before the leading edge of the next crossing pulse at the SVX. In 396ns mode, this pulse occurs in pipeline 2, 220ns before the leading edge of the next crossing pulse at the SVX. These measurements assume a 28’ grey cable to the SVX.

**Trigger Register operation**

The Trigger Register (18) puts the SVX in initialize mode for downloading, initiates Cal-Inject cycles or software-generated triggers, and sets the crossing interval for non Cal-Inject cycles. Cal-Inject mode operates in 132ns mode only.

Cal Inject Sequence long after a Preamp Reset:

<u>Command</u>	<u>Function</u>
80	Go to IDLE to prepare for Cal_inject or Downloading the SVX chip
88	Sets Acquire mode for Cal_Inject preparation
A8	(Optional) Reset the preamp before initiating a Cal_Inject
84	Initiate a Cal-Inject cycle
80	Back to IDLE

Cal Inject Sequence eighteen acquire clocks after a Preamp Reset:

<u>Command</u>	<u>Function</u>
80	Go to IDLE to prepare for Cal_inject or Downloading the SVX chip
84	Sets Acquire mode and then initiates a Cal_Inject cycle
80	Prepare for another cycle

Cal Inject Sequence with a random Preamp Reset prior to it:

<u>Command</u>	<u>Function</u>
00	Sets Acquire mode
88	Initiates a Cal-Inject cycle
80	Prepare for another cycle

To prepare for Software-generated Triggers:

<u>Command</u>	<u>Function</u>
00	Set crossing interval to 132ns, or
01	Set crossing interval to 396ns
02	Send trigger in 132ns mode, or
03	Send trigger in 396ns mode

To prepare for External Triggers:

<u>Command</u>	<u>Function</u>
00	Set crossing interval to 132ns
01	Set crossing interval to 396ns

System waits for trigger, when it occurs, BUSY goes active and an interrupt is sent to the processor. When the FIFO has been emptied of data BUSY will go low and another trigger is allowed.

## 6.4 Reading the Data

Data is read repeatedly from the Data Readout register (see section 2.3), and there are four options to determine when all the data has been read:

- 1) Read data until C0 appears twice in each byte of the readout data FIFO.
- 2) Read data until you are sure the FIFOs are empty ( $\# \text{ chips} \times 256 + 4$  bytes).
- 3) Poll bit D04 of the Trigger register while reading; it will go high when both FIFOs are empty.
- 4) Abort readout by sending a software reset.

If none of these four options is used, there will be old data in the FIFO during the subsequent readout.

This register is designed to be readable via VME Block Transfers as well as conventional reads.

## 6.5 Error Recovery

If the SVX chips become frozen, try setting the reset bit of the Control/Status register, or bit 4 of the "Shadow" register. This operation is also recommended after initial power-up of the SVX chips. These bits need not then be rewritten low.

## **7 JTAG**

The EPLDs (Erasable-Programmable Logic Devices) are programmed via Altera's "bitblaster" which plugs into the front panel of the board. The other end of the bitblaster is connected to the PC. The board's EPLDs are programmed by entering the Maxplus program, selecting the "Programmer" option, selecting the right set of files to send over the bitblaster, and clicking on the "Program" field.

### **APPENDIX A**

The Schematic Diagram for this board is number 3823-110-EC-330232 can be found in the D0 flat files at the northeast corner of the third floor of DAB.

### **APPENDIX B**

The EPLD files may be included on the following pages, but the most up-to-date versions can be found in D0server1\projects\electronics\SASeq. Use Max+plus II to view them.

**APPENDIX C**

The following is a drawing of the front panel:

